Verifying Physical Presence of Neighbors against Replay-based Attacks in Wireless Networks

Turgay Korkmaz
Department of Computer Science
The University of Texas at San Antonio
6900 North Loop 1604 West
San Antonio, TX 78249
korkmaz@cs.utsa.edu

Abstract

Verifying physical presence of a neighbor in wireless ad hoc networks is one of the key components in developing protocols resilient to replay-based attacks. For this, we first consider the RTT-based verification and revise it along with a probabilistic approach. We then consider a power-based approach and couple it with RTT-based approach to design an effective neighbor verification protocol (NVP). Using some actual experiments, we support the ideas in this paper and eliminate the impractical solutions. In theory, we always see some room for replay-based attacks. However, our proposed protocol significantly limits the effectiveness of replay-based attacks by restricting the range where they might be launched and thus makes such attacks practically impossible.

Keywords: wireless networks, neighbor discovery, wormhole attack, secure routing

I. Introduction

Wireless ad hoc networks have been receiving significant attention from research community due to their practical applications in several domains including military, industry, emergency situations, environmental monitoring and response. One of the key issues in wireless ad hoc networks is how to make underlying routing protocols more secure [1], [2]. In response to that, researchers have actually proposed various secure routing protocols using cryptographic techniques (e.g., [3], [4]). Although being resilient to various attacks, such solutions are inherently vulnerable to replay-based attacks (e.g., wormhole attacks), in which an illegitimate node overhears packets sent by some legitimate nodes and replays them within the transmission range of a different node.

For example, consider the ad hoc network in Figure 1. Assume that an attacker controls X and Y, which are connected through a dedicated channel. Now X and Y can overhear any messages transmitted by node 1 and node 6, respectively. X and Y can then exchange these messages through the dedicated channel and replay them in the transmission range of node 1 and node 6. Consequently, node 1 and node 6 consider each other as a direct neighbor

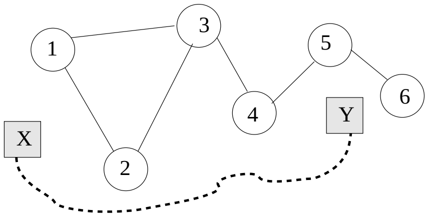

Fig. 1. A replay-based attack scenario.

Manuscript received on June 2, 2005. This work was supported by CIAS at the University of Texas at San Antonio through a grant provided by the U.S. Air Force. An abridged version of this paper was appeared in the Proceedings of the IEEE ITCC 2005, Vol. 2. pp. 704-709, Las Vegas, Nevada, April 4-6, 2005.

and thus exchange their subsequent messages through X-Y, while the messages were suppose to be exchanged through 1-3-4-5-6 in a non-adversarial environment.

By controlling the paths determined by the underlying routing protocols, the attacker can launch various attacks including denial-of-service, selective forwarding. Since the attacker does not change the content of the messages, it is not possible to deal with such attacks by just using cryptographic techniques. In addition to negatively impacting the routing protocols, replay-based intruders can affect the correctness of various other services and protocols (e.g., GPS-free positioning [5]) since the actual node locations and neighboring relations are misrepresented.

In order to detect and respond to replay-based intrusions and attacks, a general approach called packet leashes, which bound the travel distance of a packet, was introduced in [6]. However, this approach requires precise location information (e.g., using GPS) and/or tightly synchronized clocks, increasing the cost, complexity and power consumption of wireless nodes. Another approach considers the use of directional antennas [7]. However, this approach also increases the complexity, cost and power consumption. Moreover, it cannot be effective when the number of intruders is increased strategically around the target.

To better understand the underlying key issues, let us first consider how/when a reply-based attack can happen. Consider Figure 2.(a). Nodes i and j are legitimate nodes but they cannot directly communicate with each other since they are not within each other's transmission range. Intruders X and Y (one is located within node i's transmission range while the other is within that of node j) first establish a tunnel which could be a dedicated wireless or wireline channel. They then overhear the packets (e.g., HELLO messages) sent by nodes i and j, and exchange these packets through the tunnel and replay them within each other's transmission range. As a result, nodes i and j consider each other as a neighbor while they are not in reality. Note that if nodes i and j are within the transmission range of each other, as seen in Figure 2.(b), then the replay-based attacks cannot be damaging because the original packets are already received by the receiver. Moreover, replaying the already-received packets will only help legitimate nodes to detect intruders. So, replay-based attacks make sense (or are only possible) when the neighbors are not actually within each other's transmission range. Therefore, the key issue here is *how to verify whether the given two neighbors are actually within each other's transmission range or not* without increasing the complexity or requiring additional hardware (e.g,. GPS or directional antennas).

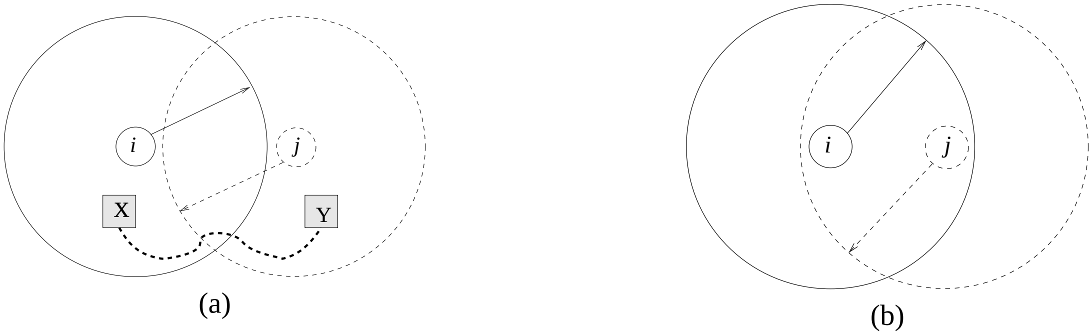

Fig. 2. How replay-based attacks happen.

If we can address this fundamental question in an efficient and scalable manner, then the replay-based attacks can easily be determined and eliminated by cancelling fake neighboring relations. For this, neighbor discover protocols (even those using cryptographic techniques) need to consider some new evidences that can *verify the physical presence of neighbor nodes*. With this in mind, researchers in [8] considered round-trip-time (RTT), that can give an upper bound on the distance between neighbor nodes. Although necessary, this bound might not be tight enough to make correct decisions in some cases, as we explain later.

Our main contributions in this paper are (i) to discuss the available techniques and challenges, (ii) to propose possible solutions with the objective of increasing the rate of making correct decisions when checking neighboring relations in wireless ad hoc networks, and (iii) to support and justify the proposed ideas, we conduct some actual experiments and analyze the collected data. Specifically, we first revise the RTT-based verification along with a probabilistic approach. We then consider the relationship between the sent and received signal powers and discuss how to benefit from such a power-based approach in providing extra evidence for the physical existence of neighbors. Subsequently, we couple the power-based approach with RTT-based approach. Our experiments also show that we should mainly rely on the RTT-based approach and use the power-based approach for double checking.

The rest of the paper is organized as follows. In Section II, we present the basic operation of the proposed neighbor verification protocol (NVP). In Section III, we discuss the basic RTT-based approach and its problems. In Section IV, we revise the RTT-approach and provide a probabilistic decision mechanism. In Section V, we discuss power-based techniques. Finally, we conclude this paper and give some directions for future research in Section VI.

II. Basic Operation of NVP

To simplify the discussion, we present the operation of the proposed neighbor verification protocol (NVP) as a stand-alone protocol. However, in practice, it can be integrated into secure neighbor discovery (SND) protocols.

Suppose nodes i and j consider each other as neighbors and want to verify if they are actually neighbors. Node i first sends RTS (Request to Send) and node j sends CTS (Clear to Send) as in IEEE 802.11 [9], [10]. If there are intruders, they can overhear these RTS/CTS frames and exchange them through the dedicated tunnel and replay them. In any case, the wireless channel is reserved (i.e., the other nodes within the transmission ranges of nodes i and j wait) for nodes i and j to exchange a data packet and its ACK. As in the previously proposed SND protocols (e.g., [8]), node i sends an authenticated message. After processing this message, node j sends an authenticated ACK. Figure 3 illustrates the main steps of the proposed protocol and the parameters that are maintained and exchanged to check whether nodes i and j can actually be neighbors.

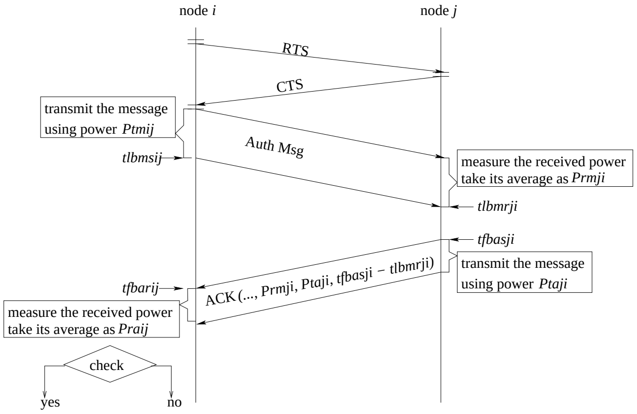
node i
node j
RTS
CTS
transmit the message using power Ptmij
tlbmsij
Auth Msg
measure the received power take its average as Prmji
tlbmrji
tfbasji
transmit the message using power Ptaji
tfbarij
ACK (..., Prmji, Ptaji, tfbasji − tlbmrji)
measure the received power take its average as Praij
check
yes
no

Fig. 3. Main steps of NVP.

Specifically, node i maintains the following variables:

- Ptm_{ij} : the power of the signal used to transmit the message to j
- $tlbms_{ij}$: the time when the last bit of the message is sent to j
- $tfbar_{ij}$: the time when the first bit of the ACK is received from j
- Pra_{ij} : the power of the received signal for the ACK from j. Note that this is an average of power measurements during the reception of the ACK rather than instantaneous ones that might be fluctuating depending on slowly varying channel conditions. For power measurements, we can use RSSI (received signal strength indicator) values that are readily available in standard wireless cards [11], [12].

Likewise, node j maintains the following variables:

- Prm_{ji} : the power of the received signal for the message from i. Again this is an average of power measurements.
- $tlbmr_{ji}$: the time when the last bit of the message is received from i
- $tfbas_{ji}$: the time when the first bit of the ACK is (will be) sent to i

- Pta_{ji} : the power of the signal used to transmit the ACK to i.

Within the ACK, node j sends Prm_{ji}, Pta_{ji}, and $proc\_delay_{ji} = tfbas_{ji} - tlbmr_{ji}$ to node i . Using these variables along with the conditions that we derive in the following sections, node i will be able to make a decision on the actuality of neighbor j.

III. Basic RTT-based Approach and Problems

Using RTT, one can compute an upper bound on the distance between nodes i and j [8]. Specifically, we consider the round trip propagation delay (RTPD), which can be computed by node i as follows.

$$RTPD_{ij} = tfbar_{ij} - tlbms_{ij} - proc\_delay_{ji}.$$

We assume that d_{ij} (the distance from i to j) and d_{ji} (the distance from j to i) are the same and simply denoted by d_{ij}. Actually, if nodes move, these values might slightly be different. However, the difference will be negligible since the speed of any wireless device is significantly less than the speed of wireless signal, i.e., nodes cannot significantly change their locations within one RTT. We can then compute an upper bound on the value of d_{ij} as

$$max\_d_{ij} = \frac{RTPD_{ij} * c}{2},$$

where c is the speed of light. Assuming that transmission ranges of nodes i and j are given as R_i and R_j, one can simply conclude that

```
Condition 1:
|    if $max\_d_{ij} \leq R_i$ and $max\_d_{ij} \leq R_j$ then
|         $i$ and $j$ are actual neighbors
|    else ???
```

In the else part of this condition, however, we cannot simply say that nodes i and j are *not* actual neighbors. This is due to the fact that max_d is not a tight bound on the actual distance since it is determined based on the speed of light which is, in general, larger than the speed of actual wireless signals [13]. In other words, within the same time (measured $RTPD$), light goes farther than the wireless signal. In addition, the errors in the measurement of $RTPD$ (even the small ones) will result in over estimating max_d. As a result, saying that nodes i and j are not actual neighbors when max_d_{ij} is greater than R_i or R_j would be wrong in some cases.

For example, consider Figure 4. If the transmission range is as shown by CASE-I, then clearly nodes i and j are actual neighbors. However, if transmission range is less than max_d (see CASE-II and CASE-III in the figure), we cannot be sure that nodes i and j are not actual neighbors. For example, as seen in the figure, they are actual neighbor in CASE-II while that is not true in CASE-III. To better deal with such cases, we revise the above condition 1 along with a probabilistic decision mechanism, as described in the next section.

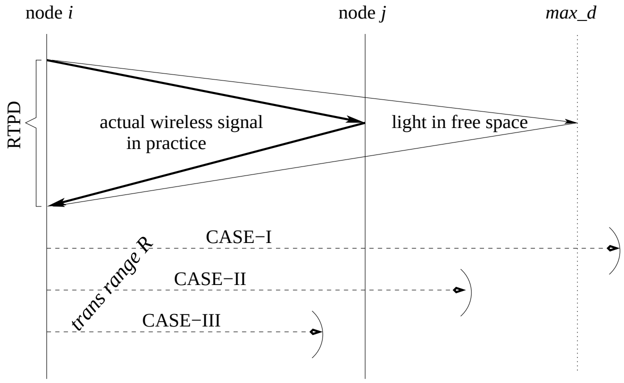

Fig. 4. Basic RTT-based approach.

IV. Revised RTT-based Approach

We first compute upper and lower bounds on the measured $RTPD_{ij}$ as follows:

$$LB = 2R/c \tag{1}$$

and

$$UB = 2R/s, \tag{2}$$

where R is the transmission range (without loss of generality, we assume that $R_i = R_j = R$), c is the speed of light, and s is the worst-case speed of wireless signal, which can be determined based on some measurements. If the measured $RTPD_{ij}$ is less than LB (or greater than UB), then nodes i and j are (or are not) actual neighbors. If $RTPD_{ij}$ is between LB and UB, then we need to consider some other decision making mechanisms. Specifically, we propose a probabilistic approach along with the following probabilistic measure:

$$\pi_{ij} = \frac{UB - RTPD_{ij}}{UB - LB} \tag{3}$$

denoting the probability that nodes i and j are actual neighbors. Using this measure, a decision maker can accept/reject neighboring relations with some level of confidence denoted by lc, $0 \leq lc \leq 1$. Accordingly, we revise the above condition 1 as follows.

```
Condition 1 (revised):
|    if $RTPD_{ij} \leq LB$ then
|      $\pi_{ij} = 1$
|    else if $RTPD_{ij} \geq UB$ then
|      $\pi_{ij} = 0$
|    else /* $LB < RTPD_{ij} < UB$ */
|      compute $\pi_{ij}$ using (3)
|    end
|    if $\pi_{ij} \geq lc$ then
|      $i$ and $j$ are actual neighbors
|    else
|      $i$ and $j$ are not actual neighbors
|    end
```

The level of confidence (lc) can be kept high or low depending on the criticality of the underlying application. However, this comes at the cost of decreasing the effective transmission range (R_{eff}) of a node or virtually increasing it. Specifically, given the level of confidence lc, we can compute the effective transmission range of a node as follows. Node i accepts node j as an actual neighbor when

$$\pi_{ij} = \frac{UB - RTPD_{ij}}{UB - LB} \geq lc,$$

In other words, nodes i and j are actual neighbors when

$$RTPD_{ij} \leq UB - lc(UB - LB).$$

Assuming the speed of wireless signal is x, $s \leq x \leq c$, then we have

$$RTPD_{ij} = 2R_{eff}/x \leq UB - lc(UB - LB).$$

Substituting (1) and (2), we can compute the maximum value for R_{eff} as

$$R_{eff} = x(1/s - lc(1/s - 1/c))R \tag{4}$$

When the speed of wireless signal x approaches to s, R_{eff} will be reduced at most

$$(lc(1 - s/c)) * 100\%$$

This simply means that some legitimate neighbors within

$$[(1 - lc(1 - s/c))R, \ \ R]$$

might wrongly be rejected. When x approaches to c, R_{eff} will be increased at most

$$((c/s - lc(c/s - 1)) - 1) * 100\%$$

This simply means that some illegitimate neighbors within

$$[R, \ \ (c/s - lc(c/s - 1))R]$$

might wrongly be accepted as actual neighbors, possibly through intruders' channel.

Clearly, there is still some room for replay-based attacks. However, such attacks are only possible when two nodes are slightly beyond each other's transmission range. If the nodes are far away from each other, replay-based attacks cannot be possible. As a result, the effectiveness of replay-based attacks is significantly reduced. For example, suppose $s = 0.9c$, $R = 100m$, and nodes i and j consider each other as neighbors. Suppose we run the proposed NVP with 90% confidence ($lc = 0.9$). In this case, a replay-based attack can only be possible if the distance between i and j is within $[100m, 101m]$. In all other cases, the proposed NVP will make the correct decision on the actuality of neighboring relations, and thus avoid replay-based attacks. Figure 5 shows the upper and lower bounds on R_{eff} under various levels of confidence. As the confidence level increases, the upper bound on R_{eff} decreases significantly. In other words, the protocol significantly reduces the scope of replay-based attacks, and thus makes them practically impossible. On the other hand, we should also note that as the confidence level increases, the lower bound on R_{eff} decreases, making protocol to reject more and more actual neighbors. For example, at the 90% confidence level, the protocol might reject actual neighbors within $[91m, 100m]$. Depending on the criticality of the underlying network mission, the decision maker can tune the lc parameter and achieve the desired level of security or coverage.

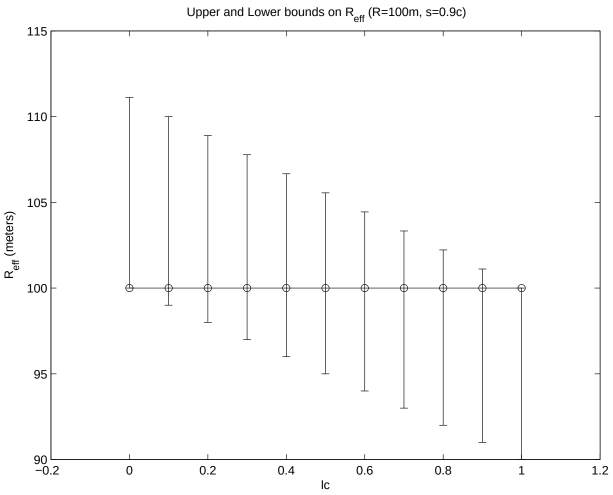

Fig. 5. Upper and lower bounds on R_{eff}.

A. Implementation and Experimental Results

The above revised RTT-based approach is easy to implement, as it involves maintaining a few variables and checking a few conditions. However, the success of this implementation depends on the precise measurements of RTPD (round trip propagation delay) which can only be done at the MAC (hardware) layer. RTT measurements at upper layers will suffer from unexpected processing delays within the operating system and between networking layers. Hence, it will be difficult to determine RTPD. As a matter of fact, we have simply implemented the proposed approach at the application layer and conducted some experiments using four (Dell Latitude D505) laptops with built-in Intel 802.11g wireless network cards. We arranged laptops as shown in Figure 6. All the laptops (nodes) operate on 100% power signal strength adjusted using Intel Pro set software. We then send 1000 packets from node 1 to each of nodes 2, 3, and 4 at different times. Accordingly, we record the RTT and take their average. In general, average RTT increases as distance increases. However, we often observed unexpected RTT values (e.g., RTT between nodes 1 and 4 is less than that between nodes 1 and 2). This is mainly due to multitasking nature of the underlying operating system (Windows XP) that causes unexpected processing delay during the measurement of RTT, and/or due to re-transmissions at the MAC layer.

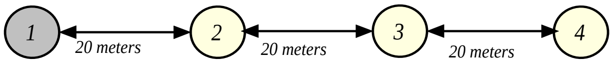

Fig. 6. A testbed for measuring RTT.

At this time, due to technical reasons, we are not able to change the MAC layer to include the proposed RTT-based approach. We expect wireless card manufacturers to update their implementation so that a wireless card can at least provide precise measurements of RTPD. In that case, our protocol can be even implemented at the application layer. Measurements about RTPD will also be useful for other services including positioning of wireless nodes. For more discussion about accurately measuring propagation delay using todays commercial, inexpensive equipment, we refer readers to [14].

V. Power-Based Approach

From a commonly used propagation model, we know that the relationship between the power of the transmitted signal (Pt) and the power of the received signal (Pr) is given by

$$Pr = \frac{K}{d^n} Pt, \tag{5}$$

where d is the distance between transmitter and receiver, K and n are the parameters determined by the characteristics of underlying wireless communications devices (e.g., antenna gains, carrier wavelength) and their surrounding areas [15], [13].

Using (5) along with the variables maintained and exchanged by nodes i and j (see Section II), we can compute d_{ij} (the distance from i to j) and d_{ji} (the distance from j to i) as follows:

$$Prm_{ji} = \frac{K}{d_{ij}^n} Ptm_{ij} \quad \Rightarrow \quad d_{ij} = \left(K \frac{Ptm_{ij}}{Prm_{ji}} \right)^{1/n}$$

and

$$Pra_{ij} = \frac{K}{d_{ji}^n} Pta_{ji} \quad \Rightarrow \quad d_{ji} = \left(K \frac{Pta_{ji}}{Pra_{ij}} \right)^{1/n}$$

It seems that one can simply conclude that nodes i and j are actual neighbors if $d_{ij} \leq R_i$ and $d_{ji} \leq R_j$. Unfortunately, we have two key issues that hinder such a conclusion.

1) The values of K and n vary and possibly not known due to different conditions in different surrounding areas. Therefore, it is hard to compute the exact distance [16].

2) The intruders can increase the received power by adjusting the power of their replayed signal so that the distance will appear smaller than it is suppose to be.

To address these issues, respectively, we propose to:

1) Avoid the need for knowing the explicit values of varying parameters K or n.
2) Couple power-based approach with previously discussed RTT-based approaches.

The challenging question is now how to achieve these objectives and how to benefit from them in the design of a neighbor verification protocol.

A. Avoiding Explicit Values of K and n and Coupling RTT- and Power-based Approaches

We assume that the values of d_{ij} and d_{ji} should be the same (the difference in case of node movements will be negligible since the speed of wireless signal is significantly larger than the speed of any wireless device, so within one RTT, a node cannot significantly change its location). In addition, environmental conditions that affect the values of K and n are not likely to change significantly within one RTT. So we can assume that the values of K and n (albeit slowly varying in general) are relatively stable during the exchange of a message and its ACK so that they are assumed to be the same from the view point of the average received power at node i and j within one RTT. As a result, we can avoid the need to know the explicit values of K or n and conclude that

$$\frac{Ptm_{ij}}{Prm_{ji}} \cong \frac{Pta_{ji}}{Pra_{ij}} \tag{6}$$

for actual neighbors, where $\cong$ means 'is equal (or close enough) to'.

For this case we also conducted actual experiments, where we used two (Dell Latitude D505) laptops with built-in Intel 802.11g wireless network cards. We arranged laptops as shown in Figure 7. We considered three different levels of transmission power 100%, 65%, and 25% using the Intel(R) ProSet utility (both nodes use the same power level). We then send 1000 packets from node i to each of node j and its ACK from j to i. Accordingly, we record the received signal level at nodes i and j for each packet and its ACK. Since the transmission power level is the same, we can re-arrange the condition 6 and expect that

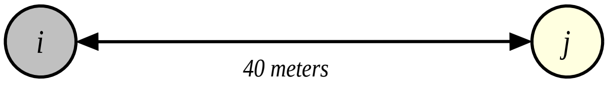

Fig. 7. A testbed for measuring received power strength.

$$\frac{Pra_{ij}}{Prm_{ji}} \cong \frac{Pta_{ji}}{Ptm_{ij}} \cong 1. \tag{7}$$

Due randomness in the channel, the received power strengths fluctuate from one packet to another even we do not move laptops at all. Figure 8 shows the measured received power strength for the sent packets and received ACKs. Despite the fluctuation, the ratio in 7 seems to be very close to one, as shown in Figure 9. These experimental results indicate that even though the randomness in the channel affects the received signal strength at different times, it can be assumed relatively constant within one RTT, as we considered above.

If nodes are not actual neighbors and thus communicate through intruders, then the values of Prm_{ji} and Pra_{ij} will be altered depending on the transmission power used during replaying messages and the distance between the intruders and legitimate nodes. If this alteration is done arbitrarily, then $\frac{Ptm_{ij}}{Prm_{ji}}$ and $\frac{Pta_{ji}}{Pra_{ij}}$ cannot be close enough. In that case, we can be sure that nodes i and j are not actual neighbors. When (6) is satisfied, we need to be careful because it might be possible for well-equipped intruders to make this condition to hold as follows.

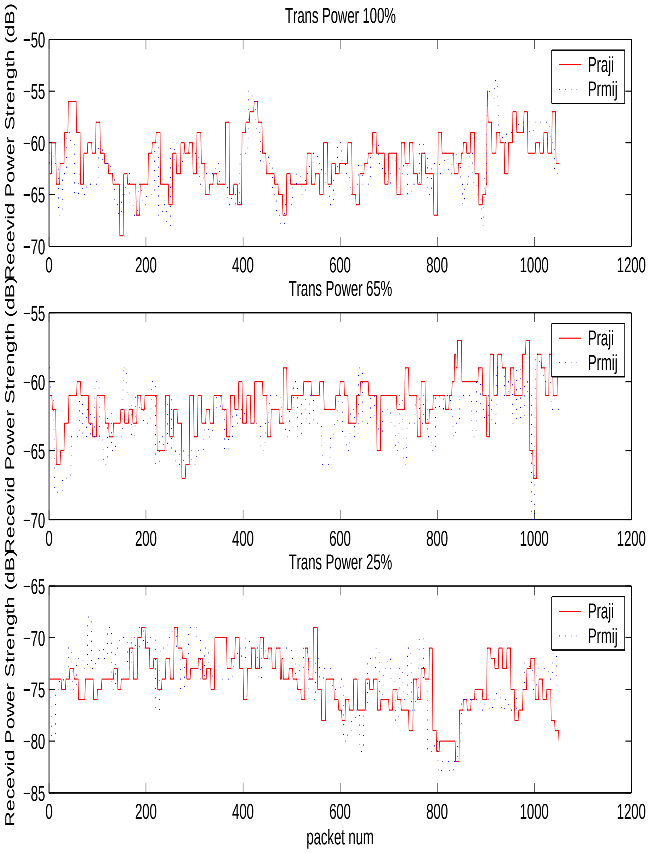

Fig. 8. Received power strengths per packet and its ACK.

Suppose the distance from node i to intruder X is d_{ix} and from node j to intruder Y is d_{jy}. Then intruder X receives the message from i with the power of $Prm_{xi} = \frac{K_{ix}}{d_{ix}^n} Ptm_{ij}$. If X asks Y to transmit this message using power $Ptm_{yj} = \beta Prm_{xi}$, then the received power at node j will be

$$Prm_{ji} = \beta \frac{K_{jy}}{d_{jy}^n} \frac{K_{ix}}{d_{ix}^n} Ptm_{ij}$$

Similarly, if Y asks X to transmit the ACK using $Pta_{xi} = \beta Pra_{yj}$, then the received power at node i will be

$$Pra_{ij} = \beta \frac{K_{ix}}{d_{ix}^n} \frac{K_{jy}}{d_{jy}^n} Pta_{ji}$$

Clearly, these values of Prm_{ji} and Pra_{ij} seem to satisfy the condition (6) even though nodes i and j are not actual neighbors. However, in practice, due to randomness in K, attackers may not tune their parameters to satisfy the condition (6. Nevertheless, in [17], we proposed coupling RTT-based approach with power-based approach using the following relations

$$\frac{Pr}{Pt} = \frac{K}{d^n} \text{ and } d = \frac{RTPD * x}{2},$$

where x is the speed of wireless signal. From these two, it seems that the ratio $\frac{Pt}{Pr}$ in (6) is actually a function of $RTPD$. If we know or can characterize this function (e.g., using linear regression [18]), we can impose an additional condition to check the actuality of a neighbor. However, based on our experimental results shown in Figure 10, we realized that there is no strong relationship between the value of P_r/P_t and measured RTT. This

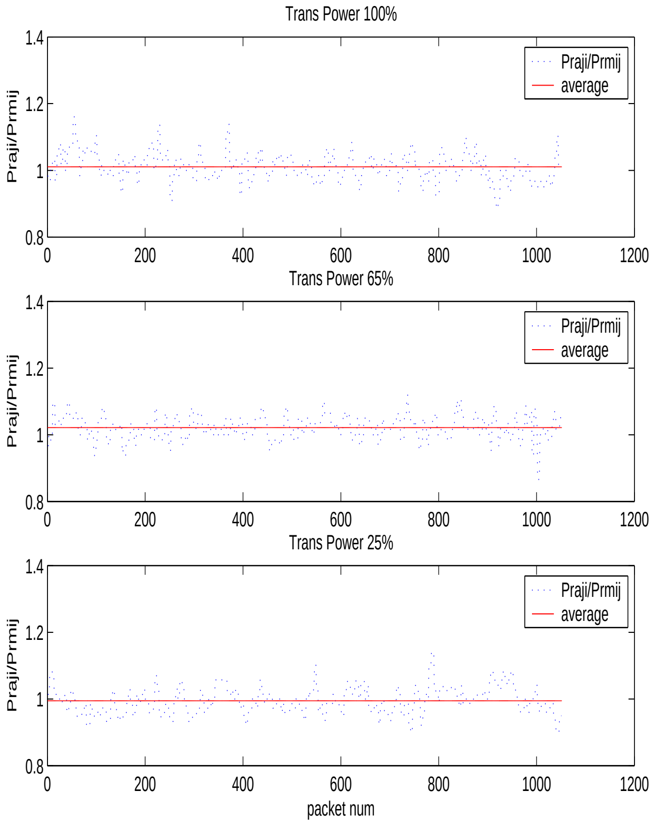

Fig. 9. Ratio of received power strengths.

is mainly due to the randomness in K which changes from one packet to another since they are transmitted at different times.

In summary, we can look at the equality of the ratios in (6) and use it to verify the actuality of neighbors. However, in some cases, this equality could hold even the neighbors are not actual. Therefore, the power-based approach should be used along with RTT-based approaches as a double check rather than the main check.

VI. Conclusions and Future Work

We first studied how to verify the existence of a physical neighbor using a RTT-based approach. In contrast to previously proposed version, we determined lower and upper bounds on the measured $RTPD$ and proposed a probabilistic decision mechanism when the $RTPD$ is between the lower and upper bounds. To provide additional evidences for the physical presence of neighboring nodes, we also discussed how to use the relationship between the sent and received signals and couple it with RTT-based approaches. In general, it is not possible (at least in theory) to provide 100% protection against replay-based attacks. However, by combining RTT- and power-based mechanisms along with cryptographic techniques, the proposed NVP provides a tunable neighbor verification against replay-based attacks and makes such attacks practically impossible by limiting the scop where such attacks can be lunched.

Due to technical difficulties in making modifications at the MAC layer, we were not able to fully implement the proposed protocol at the the MAC layer. However, we conducted several experiments to support the ideas presented in the paper. One important issue is the accuracy of the measured RTPD. Due to multitasking and processing overheads, it is not possible to get good estimates for this measures by using application layer implementations. To provide precise measurements of RTPD, we expect wireless card manufacturers to update their implementation

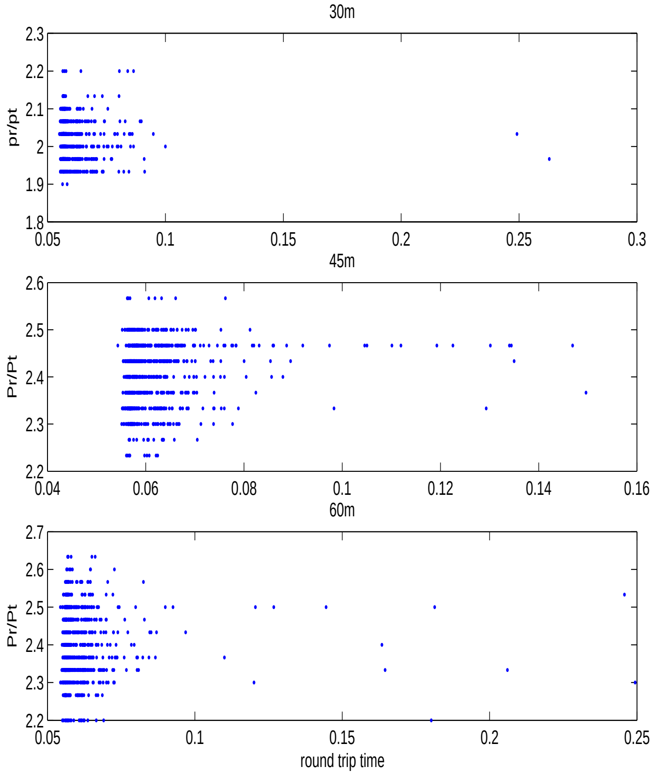

Fig. 10. Ratio of received power strengths.

as such measurements are also necessary for other services including positioning of wireless nodes. Upon the availability of such measure, we can even implement the proposed protocol at the application layer.

In addition to RTT- and power-based approaches, we plan to consider new mechanisms as the future work. For example, we will focus on RTS/CTS patterns in a given neighborhood and analyze them with the objective of detecting physically impossible neighboring relations. We will further extend our testbed and conduct more actual experiments with the objective of developing the practically possible solutions.

VII. Acknowledgment

This work was supported by CIAS at the University of Texas at San Antonio through a grant provided by the U.S. Air Force. The author also thanks Prashant Reddy Gurijala, a master student at UTSA, for his help in developing the required programs using .NET framework and conducting the actual experiments.

References

[1] L. Zhou and Z. Haas, "Securing ad hoc networks," *IEEE Network*, vol. 13, no. 6, pp. 24–30, Nov.-Dec. 1999.

[2] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures," in *Proceedings of the First Sensor Network Protocols and Applications*. IEEE, 11 May 2003, pp. 113 – 127.

[3] K. Sanzgiri, B. Dahill, B. Levine, C. Shields, and E. Belding-Royer, "A secure routing protocol for ad hoc networks," in *Proceedings of the 10th IEEE International Conference on Network Protocols*. IEEE, 12-15 Nov. 2002, pp. 78 – 87.

[4] W. Zhang, R. Rao, G. Cao, and G. Kesidis, "Secure routing in ad hoc networks and a related intrusion detection problem," in *MILCOM 2003*, vol. 2. IEEE, 13-16 Oct. 2003, pp. 735 – 740.

[5] S. Capkun, M. Hamdi, and J.-P. Hubaux, "GPS-free positioning in mobile ad-hoc networks," in *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*. IEEE, 3-6 Jan. 2001, pp. 3481 – 3490.

[6] Y.-C. Hu, A. Perrig, and D. Johnson, "Packet leashes: a defense against wormhole attacks in wireless networks," in *Proceedings of the INFOCOM 2003 Conference*, vol. 3. IEEE, 30 March - 3 April 2003, pp. 1976 – 1986.

[7] L. Hu and D. Evans, “Using directional antennas to prevent wormhole attacks,” in *The 11th Annual Network and Distributed System Security Symposium (NDSS ’04)*, 5-6 Feb. 2004.

[8] Y.-C. Hu, A. Perrig, and D. Johnson, “Rushing attacks and defense in wireless ad hoc network routing protocols,” in *Proceedings of the 2003 ACM workshop on Wireless security*. ACM, 2003, pp. 30 – 40.

[9] *IEEE 802.11 General Info*,
http://grouper.ieee.org/groups/802/11/main.html.

[10] J. Schiller, *Mobile Communications*, 2nd ed. Addison-Wesley, 2004.

[11] J. Bardwell, “Converting signal strength percentage to dBm values,” WildPackets, Tech. Rep., Nov. 2002,
www.wildpackets.com/elements/whitepapers/Converting_Signal_Strength.pdf.

[12] *Intel PRO/Wireless 2200BG*,
http://support.intel.com/support/wireless/wlan/.

[13] W. Stallings, *Data and Computer Communications*, 7th ed. Prentice Hall, 2004.

[14] A. Gunther and C. Hoene, “Measuring round trip times to determine the distance between wlan nodes,” in *Proceedings of the Networking 2005*, Waterloo, Canada, May 2005 2005, (to appear).

[15] S. Haykin, *Communication Systems*, 4th ed. Wiley, 2001.

[16] A. Savvides, C.-C. Han, and M. B. Strivastava, “Dynamic fine-grained localization in ad-hoc networks of sensors,” in *Proceedings of the 7th annual international conference on Mobile computing and networking*. ACM, 2001, pp. 166 – 179.

[17] T. Korkmaz, “Verifying physical presence of neighbors against replay-based attacks in wireless ad hoc networks,” in *Proceedings of the ITCC 2005 Conference*, vol. 2. Las vegas, NV: IEEE, April 4-6 2005, pp. 704–709.

[18] A. M. Law and W. D. Kelton, *Simulation Modeling & Analysis*, 2nd ed. McGraw-Hill book company, 1991.

Turgay Korkmaz received the B.Sc. degree with the first ranking from Computer Science and Engineering at Hacettepe University, Ankara, Turkey, in 1994, and two M.Sc. degrees from Computer Engineering at Bilkent University, Ankara, and Computer and Information Science at Syracuse University, Syracuse, NY, in 1996 and 1997, respectively. In Dec 2001, Dr. Korkmaz received his PhD degree from Elec. and Computer Eng. at University of Arizona, under the supervision of Dr. Marwan Krunz. In January 2002, he joined the University of Texas at San Antonio, where he is currently an Assistant Professor of Computer Science department.

Dr. Korkmaz’s research interests include QoS-based routing, multiple constrained path selection, efficient dissemination of network-state information in converging networks. He is also interested in wireless networks and security issues.

Dr. Korkmaz was a Co-PI on the NSF High Performance Network Connections (HPNC) Award to provide Internet 2 Connectivity for UTHSCSA and UTSA. He was the co-chair for the ACM Symposium on Applied Computing (SAC ’03), Special Track on Parallel and Distributed Systems and Networking; and the SAC ’04 Special Track on Computer networks. He has been serving on the technical program committee of IEEE INFOCOM 2004, 2005, and 2006.